

IN THE CLAIMS

1. (**Currently Amended**) A method for handling runtime objects operating on a wireless device, wherein the runtime objects are instantiations of classes, wherein a class can implement an object-oriented interface having methods, comprising the steps of :

assigning, for a class, an interface ordinal to an interface based upon a relationship existing between the class and the interface;

associating with at least one interface ordinal a reference that points to an interface implemented by a class, thereby generating a class interface check table for the class, wherein more than one interface may share the same interface ordinal in order to minimize the number of interface ordinals;

assigning, for a class, a method ordinal to an interface's method based upon a relationship existing between the class and the method, wherein more than one method may share the same method ordinal in order to minimize the number of method ordinals;

associating with at least one method ordinal a reference that points to a method of an interface, thereby generating a class interface dispatch table for the class, wherein the generated class interface dispatch table is sparse and;

compacting class interface dispatch tables of a plurality of classes, thereby generating a compact dispatch table, by overlaying the sparse class interface dispatch tables so that non-null entries of the class interface dispatch tables at ordinal position fit into the compact dispatch table;

wherein an assigned interface ordinal and an associated reference stored in the class interface check table are to be used to check whether a runtime object operating on the wireless device is an instantiation of an interface and;

wherein an assigned method ordinal and an associated method reference stored in the compact dispatch table are to be used to perform a dispatch of an interface's method implemented by the runtime object.

2. **(Currently Amended)** The method of claim 1, wherein a direct interface check is performed ~~to verify if the runtime object can be cast into the type of an interface~~ by performing steps comprising:

providing class data for the runtime object in order to determine what class type provided the constructor for the runtime object and;

using the interface ordinals' association with the interface references to check whether the runtime object is an instance of an interface;

wherein the interface allows multiple inheritance to occur without multiple inheritance of implementation.

3. **(Original)** The method of claim 1, wherein a dispatch is performed by performing steps comprising:

providing class data for the runtime object in order to determine what class type provided the constructor for the runtime object;

using the method ordinals association with the method references to directly dispatch the method.

4. **(Original)** The method of claim 1, wherein a compiler system generates check and dispatch information based upon an input set of classes and set of interfaces,

wherein the check and dispatch information includes the interface ordinals, the interface references associated with the interface ordinals, the method ordinals, and the method references associated with the method ordinals, wherein the wireless device includes runtime storage to store on the wireless device the assigned interface ordinals and their associated interface references.

5. **(Original)** The method of claim 4, wherein the runtime storage stores on the wireless device the assigned method ordinals and their associated method references.

6. **(Original)** The method of claim 5, wherein the runtime storage comprises runtime context which includes objects for which a direct check is performed and includes method calls on objects for which a direct dispatch is performed.

7. **(Original)** The method of claim 6, wherein the wireless device includes a runtime processor and operates within a Java-based computer environment, wherein an assigned interface ordinal and an associated reference are to be used by the runtime processor to check whether a runtime object operating on the wireless device is an instantiation of an interface;

wherein an assigned method ordinal and an associated method reference are to be used by the runtime processor to perform a dispatch of an interface's method implemented by the runtime object.

8. **(Original)** The method of claim 7, wherein a method call on the wireless device is used as a trigger to invoke a direct check and direct dispatch using the check and dispatch information.

9. **(Currently Amended)** A system for handling runtime objects operating on a wireless device, wherein the runtime objects are instantiations of classes, wherein a class can implement an object-oriented interface having methods, comprising:

a computer-readable storage medium for storing a class interface check data structure,
wherein the class interface check data structure stores associations between interface ordinals and references to interfaces,

wherein the ordinals ~~were~~are assigned to interfaces such that a first constraint is substantially satisfied, wherein the first constraint provides that two interfaces are assigned different interface ordinals if there exists a class that implements both of the interfaces;

said computer-readable storage medium storing a compact class interface dispatch data structure, wherein the compact class interface dispatch data structure stores associations between method ordinals and references to methods,

wherein the ordinals ~~were~~are assigned to methods such that a second constraint ~~was~~is substantially satisfied, wherein the second constraint provides that two methods have different method ordinals if there exists a class that implements both of the methods;

wherein a compact class interface dispatch data structure is generated by overlaying the class interface dispatch data structures of a plurality of classes, each storing associations between method ordinals and references to methods implemented by the respective class, so that non-null entries of the class interface dispatch data structures at ordinal position fit into the compact class interface dispatch data structure;

wherein the class interface check data structure is to be used to check whether a runtime object operating on the wireless device is an instantiation of an interface;

wherein the compact class interface dispatch data structure is to be used to perform a dispatch of an interface's method implemented by the runtime object.

10. **(Original)** The system of claim 9 further comprising:

a compiler system that assigns the ordinals to interfaces such that the first constraint is substantially satisfied, wherein the compiler assigns the ordinals to method such that the second constraint is substantially satisfied.